Announcements

- Homework #3 due tonight at 11:59pm
- Homework #4 out now
\{ \text{true} \}

x := m;
y := 0;
\textbf{while} (n < x) \textbf{do} (  
  x := x - n;
  y := y + 1
)

\{ \}
\{\text{true}\}

\text{x := m;}
\text{y := 0;}
\text{while (n < x) do (}
\text{  x := x - n;}
\text{  y := y + 1}
\text{)}

\{\text{n \times y + x = m}\}

In other words, the program divides m by n, so y is the quotient and x is the remainder.
Generating Preconditions

To fill in a precondition:

\[ \{ \quad \} \ c \ \{ Q \} \]

there are many possible preconditions—and some are more useful than others.
Weakest Preconditions

**Intuition:** The weakest liberal precondition for $c$ and $Q$ is the weakest assertion $P$ such that $\{P\} \ c \ \{Q\}$ is valid.
Weakest Preconditions

**Intuition:** The weakest liberal precondition for $c$ and $Q$ is the weakest assertion $P$ such that $\{P\} c \{Q\}$ is valid.

More formally...

**Definition (Weakest Liberal Precondition)**

$P$ is a weakest liberal precondition of $c$ and $Q$ written $\text{wlp}(c, Q)$ if:

$$\forall \sigma, I. \sigma \models_I P \iff (C[c] \sigma) \text{ undefined } \lor (C[c] \sigma) \models_I Q$$
Weakest Preconditions

\[ wlp(\text{skip}, P) = P \]
Weakest Preconditions

\[
wlp(\text{skip}, P) = P \\
wlp(x := a, P) = P[a/x]
\]
Weakest Preconditions

\[
\begin{align*}
\text{wlp}(\text{skip}, P) &= P \\
\text{wlp}(x := a, P) &= P[a/x] \\
\text{wlp}((c_1; c_2), P) &= \text{wlp}(c_1, \text{wlp}(c_2, P))
\end{align*}
\]
Weakest Preconditions

\[
\begin{align*}
\text{wlp}(\textbf{skip}, P) &= P \\
\text{wlp}(x := a, P) &= P[a/x] \\
\text{wlp}((c_1; c_2), P) &= \text{wlp}(c_1, \text{wlp}(c_2, P)) \\
\text{wlp(\textbf{if} b \textbf{ then} c_1 \textbf{ else} c_2, P)} &= (b \implies \text{wlp}(c_1, P)) \land (\neg b \implies \text{wlp}(c_2, P))
\end{align*}
\]
Weakest Preconditions

\[
\begin{align*}
  \text{wlp}(\text{skip}, P) &= P \\
  \text{wlp}(x := a, P) &= P[a/x] \\
  \text{wlp}((c_1; c_2), P) &= \text{wlp}(c_1, \text{wlp}(c_2, P)) \\
  \text{wlp}(\text{if } b \text{ then } c_1 \text{ else } c_2, P) &= (b \implies \text{wlp}(c_1, P)) \land \\
  & \quad (\neg b \implies \text{wlp}(c_2, P)) \\
  \text{wlp}(\text{while } b \text{ do } c, P) &= \bigwedge_i F_i(P)
\end{align*}
\]
Weakest Preconditions

\[
\begin{align*}
\text{wlp(} \text{skip, } P) & = P \\
\text{wlp(} x := a, P) & = P[a/x] \\
\text{wlp(} (c_1; c_2), P) & = \text{wlp}(c_1, \text{wlp}(c_2, P)) \\
\text{wlp(} \text{if } b \text{ then } c_1 \text{ else } c_2, P) & = (b \implies \text{wlp}(c_1, P)) \land (\neg b \implies \text{wlp}(c_2, P)) \\
\text{wlp(} \text{while } b \text{ do } c, P) & = \bigwedge_i F_i(P)
\end{align*}
\]

where

\[
\begin{align*}
F_0(P) & = \text{true} \\
F_{i+1}(P) & = (\neg b \implies P) \land (b \implies \text{wlp}(c, F_i(P)))
\end{align*}
\]
Applications of Weakest Preconditions

Failing fast: avoid wasting work on bad inputs.

\[ \text{p := getPacket();} \]
\[ \text{processPacket(p);} \]
\[ \text{assert} \ P_{\text{safe}} \]
Applications of Weakest Preconditions

Failing fast: avoid wasting work on bad inputs.

\[ p := \text{getPacket}(); \]
\[ \text{processPacket}(p); \]
\[ \{ P_{\text{safe}} \} \]
Applications of Weakest Preconditions

Failing fast: avoid wasting work on bad inputs.

\[ p := \text{getPacket}(); \]
\[ \{ P_{\text{filter}}(p) \}; \]
\[ \text{processPacket}(p); \]
\[ \{ P_{\text{safe}} \} \]
Applications of Weakest Preconditions

Failing fast: avoid wasting work on bad inputs.

\[ p := \text{getPacket}(); \]
\[ \textbf{assert} \ P_{\text{filter}}(p); \]
\[ \text{processPacket}(p); \]
Applications of Weakest Preconditions

Failing fast: avoid wasting work on bad inputs.

\[ p := \text{getPacket}(); \]
\[ \textbf{assert} \; P_{\text{filter}}(p); \]
\[ \text{processPacket}(p); \]

\( P_{\text{filter}} \) should be the \textit{weakest} precondition to avoid ruling out legitimate inputs.

Properties of Weakest Preconditions

Lemma (Correctness of Weakest Preconditions)

\[ \forall c \in \text{Com}, Q \in \text{Assn}. \]
\[ \vdash \{ wlp(c, Q) \} \ c \ \{ Q \} \ \text{and} \]
\[ \forall R \in \text{Assn.} \ \vdash \{ R \} \ c \ \{ Q \} \ \text{implies} \ (R \ \implies \ wlp(c, Q)) \]
Properties of Weakest Preconditions

Lemma (Correctness of Weakest Preconditions)

\[ \forall c \in \text{Com}, Q \in \text{Assn}. \]
\[ \vdash \{ wlp(c, Q) \} c \{ Q \} \quad \text{and} \]
\[ \forall R \in \text{Assn}. \vdash \{ R \} c \{ Q \} \text{ implies } (R \implies wlp(c, Q)) \]

Lemma (Provability of Weakest Preconditions)

\[ \forall c \in \text{Com}, Q \in \text{Assn}. \vdash \{ wlp(c, Q) \} c \{ Q \} \]
Soundness and Completeness

**Soundness:** If we can prove it, then it’s actually true.

**Completeness:** If it’s true, then a proof exists.
Soundness and Completeness

**Soundness**: If we can prove it, then it’s actually true.

**Definition (Soundness)**

\[
\text{If } \vdash \{P\} \text{ c } \{Q\} \text{ then } \models \{P\} \text{ c } \{Q\}.
\]

**Completeness**: If it’s true, then a proof exists.

**Definition (Completeness)**

\[
\text{If } \models \{P\} \text{ c } \{Q\} \text{ then } \vdash \{P\} \text{ c } \{Q\}.
\]
Kurt Gödel vs. Sir Tony Hoare
Relative Completeness

**Theorem (Cook (1974))**

\[ \forall P, Q \in \text{Assn}, c \in \text{Com}. \vdash \{P\} c \{Q\} \implies \vdash \{P\} c \{Q\}. \]
Relative Completeness

**Theorem (Cook (1974))**

\[ \forall P, Q \in \textbf{Assn}, c \in \textbf{Com}. \models \{P\} c \{Q\} \text{ implies } \vdash \{P\} c \{Q\}. \]

**Proof Sketch.**

Let \(\{P\} c \{Q\}\) be a valid partial correctness specification. By the first Lemma we have \(\models P \implies \text{wlp}(c, Q)\).

By the second Lemma we have \(\vdash \{\text{wlp}(c, Q)\} c \{Q\}\).

We conclude \(\vdash \{P\} c \{Q\}\) using the CONSEQUENCE rule.